**REMARKS**

Claims 1 – 15 were presented for examination and all were rejected under 35 U.S.C. § 103. Claims 1 – 15 are being resubmitted without amendments. In view of following remarks, withdrawal of the rejections of claims 1 – 15 is respectfully requested.

CLAIM REJECTIONS UNDER 35 U.S.C. § 103 – Eranian and Gillies

In paragraph 2 of the above-mentioned Final Office Action, claim 1 was rejected under 35 U.S.C. § 103 (a) as being unpatentable over Eranian et al., "The Making of Linux/ia64," Internet Systems and Applications Laboratory, August 1999, pp. 1-11 ("Eranian") in view of USPubN: 2002/0083425 by Gillies ("Gillies"). The rejection is traversed. Eranian and Gillies, either alone or in combination, do not disclose every element of the claimed invention. The alleged motivation for combining the teachings of Eranian and Gillies is improper. Showing a prima facie case of obviousness failed.

In brief, the claimed invention is about allocating N number of registers for use in code instrumentation of a block of programming code. In effect, claim 1 recites *a first statement* and *a second statement* to allocate registers. Further, the second statement includes second parameters that are generated by using *the number N* of registers to be allocated and the *first parameters used in the first statement* as inputs. In the illustrative embodiment of FIG. 4, the first statement is alloc (. . ., 3, 5, 2, . . .) and the second statement is alloc (. . ., 3, 5, 6, . . . ). Alternatively, the second statement may be expressed as alloc (. . ., 3, 5, 2 + 4, . . .) wherein N=4.

In contrast, the cited Figure 5 of Eranian shows *only one* alloc statement, which is B:alloc r32=ar.pfs, 5, 4, 0, 0 and which is corresponded to the claimed second statement. Eranian *does not disclose* any statement corresponding to the

claimed first statement allocating registers. In fact, the Final Office Action *did not cite* any statement in Eranina that corresponds to this claimed first statement, and for the claimed element "identifying a first statement allocating registers, the first statement is associated with a block of programming code," the Final Office Action cited only function A, Figure 5, and "allocating registers for a function implicitly discloses a block of related programming code." Even for the sake of argument that Eranian's function A corresponds to the claimed block of programming code, Eranian still misses the claimed element "*identifying a first statement allocating register*" (emphasis added).

Because Eranian does not disclose the first statement, Eranian cannot disclose the first *parameters used in the first statement*, and thus cannot disclose using *the first parameters (used in the first statement) as inputs* in generating the second parameters to allocate the N number of registers. Like above, assume that the cited function A corresponds to the claimed block of programming code, because Eranian does not disclose the first statement, Eranian cannot disclose the first statement being associated with a block of programming code. To be parallel with claim 1, Eranian must, but *does not*, disclose an alloc statement associated with function A, assuming the statement B:alloc corresponds to the claimed second statement. While Eranian disclose B:alloc r32=ar.pfs, 5, 4, 0, 0, there is nothing in Eranian to indicate that 5, 4, 0, 0 is based on first parameters *used in the first statement*, e.g., used in another alloc statement. Those skilled in the art will recognize that all r32=ar.pfs indicates that previous function is stored in r32. The Office Action asserted that "reuse of r32-r37 of statement A is using input of N number of registers and r32-41 amounts to second parameters being generated." It is respectfully submitted that it is not clear what is meant by "reuse of r32-r37 of statement A is using input of N number of registers and r32-r41 amounts to second parameters being generated" and how this assertion is

relevant to the claimed invention. Even though Eranian shows function A associated with r32, r37, etc., this function A with r32, r37 is *not a statement allocating registers.* "Alloc" is statement allocating registers.

Because Eranian does not disclose every element of claim 1, and Gillies does not provide the elements missing in Eranian, Eranina and Gillies, either alone or in combination, do not anticipate claim 1.

The Office Action conceded that Eranian does not disclose allocating N number of registers is *for use in code instrumentation of the block of programming code.* The Office Action then concluded that "[b]ut Eranian discloses debug and simulation . . . ., hence has suggested instrumentation of programming code, a concept so well known in the software development and code simulation and testing." While it is true that instrumentation of programming code is well known in software, code instrumentation may be used for many purposes other than simulation, debug, and testing. Therefore, reciting that disclosing debug and simulation has suggested instrumentation code is improper as it is a conclusory statement without supporting evidence and impermissibly relies on hindsight.

The Office Action also concluded that "[a]nalogous to the modification of register input to provide increased output registers as by Eranian, Gillies discloses modification of the number of original registers are allowing additional registers to be available for code instrumentation (e.g., Fig. 2-4, 8-9). It would have been obvious . . . to implement the register allocation techniques by Eranian so that it be used for code instrumentation as taught by Gillies . . ." It is respectfully submitted that while it is true that Gillies teaches allocating registers, again the Office Action's assertion is at best a conclusory statement relying on hindsight. There is nothing in either Eranian or Gillies that suggests combination of the two teachings. The alleged motivation for

combining Eranian and Gillies is improper, and showing a prima facie case of obviousness fails.

For the foregoing reasons, claim 1 is patentable. Claims 2-5 depend from claim 1, and are therefore patentable for at least the same reasons as claim 1. Claims 2-5 are also patentable for their additional limitations.

Regarding claim 3 with the further limitation "modifying the number O of the first parameters to generate the number O of the second parameters," the Office Action cited Figure 5 and *asserted without explaining* "outputs from function A and combined output from B." Those skilled in the art will recognize that alloc . . . 5, 4, 0, 0 of Eranian indicates that *zero* register is to be allocated for the output registers. This is evidenced by the fact that only input and local registers, but *no output registers*, are shown associated with the B: alloc statement. Further, there is nothing in Eranian that indicates modifying the *number O (of output register)* of the first parameters to generate the number O of output registers of the second parameters. Additionally, as indicated above, there is no first statement and no first parameters in Eranian, there cannot be modifying the first number O of output registers used in the first statement.

Regarding claim 4, the Office Action asserted "function A call *output parameters* r47-r52 matching with *Input parameters* r32-r36 of the Output of the B call is equivalent to using the O of the first parameters, while r37-r41 in B output is the equivalent of N from the 1st parameters so that O from B is the combined O and N from the 1st parameter." It is respectfully submitted that the recitation of claim 4 is not about "using the O of the first parameters" or using output parameters as Input parameter as cited by the Office Action. Rather, reading in conjunction with claim 1 and 2, claim 4 recites using the *number O of output registers* of the first parameters in generating *the number O of output registers* of the second parameters. In the

illustrative embodiment of FIG. 4, the number O of output registers of the first parameters is 2 while the number O of output registers of the second parameters is 6, and N equals to 4, which indicates that 4 registers is to be allocated. Additionally, the number N (or 4 in the illustrative embodiment) is not "from the 1st parameters," as asserted by the Office Action. In fact, the number N is the number of registers to be allocated. Further, as indicated above with regards to claim 3, there is no first statement and no first parameters in Eranian, and, as a result, there cannot be using the number O of output registers of the first parameter as inputs in generating the number O of the second parameter. Further, as also explained in connection with claim 3, in Eranian, *zero* number of output registers is to be allocated from the statement alloc . . . 5, 4, 0, 0, and there is no indication that this zero number of output registers to be allocated is a result of using the N (or 4) of registers to be allocated and the number O of output registers of the first parameters used in the first statement.

Explanation with respect to claim 4 is applicable to claim 5. Further, as indicated, the number O of output registers in the B: alloc statement is *zero* (alloc 5, 4, *0*, 0), and there is no indication that this *zero equals* the number N of registers to be allocated and the number O of output registers of the fist parameters used in the first statement.

Therefore, claims 3-5 are patentable for their additional limitations.

Claims 6-10 and 11-15 recite limitations corresponding to claims 1-5, and are therefore patentable for at least the same reasons as claims 1-5.

**SUMMARY**

In conclusion, it is submitted that pending claims clearly present subject matter that is patentable over the prior art of record, and withdrawal of the rejections is respectfully solicited.

Respectfully submitted,

Date: 1/10/05 By: _____

Tuan V. Ngo, Reg. No. 44,259
IP Administration
Legal Department, M/S 35
Hewlett-Packard Company
P. O. Box 272400
Fort Collins, CO 80527-2400
Phone (408) 447-8133
Fax (408) 447-0854